Experimental Evaluation for IPv6 over VANET Geographic routing

Manabu Tsukada
INRIA Paris - Rocquencourt
Domaine de Voluceau, BP 105
78153, Le Chesnay, France
Manabu.Tsukada@inria.fr

Ines Ben Jemaa
INRIA Paris - Rocquencourt
Domaine de Voluceau, BP 105
78153, Le Chesnay, France
Ines.Benjemaa@inria.fr

Hamid Menouar
HITACHI Europe SAS
955 route des Lucioles
Sophia Antipolis, France
Hamid.Menouar@hitachi-eu.com

Wenhui Zhang
NEC Europe Ltd.
NEC Laboratories Europe, Network Division
Kurfürsten-Anlage 36, 69115
Heidelberg, Germany
Wenhui.Zhang@nw.neclab.eu

Maria Goleva
NEC Europe Ltd.
NEC Laboratories Europe, Network Division
Kurfürsten-Anlage 36, 69115
Heidelberg, Germany
Maria.Goleva@nw.neclab.eu

Thierry Ernst
INRIA Paris - Rocquencourt
Domaine de Voluceau, BP 105
78153, Le Chesnay, France
Thierry.Ernst@inria.fr

ABSTRACT

Vehicular communication is an important part of the Intelligent Transportation Systems (ITS). Geographic routing in vehicular ad hoc network (VANET) is becoming an interesting topic to deliver safety messages between cars but also between a car and a roadside infrastructure within a designated destination area. The Car2Car Communication Consortium specified C2CNet architecture as a geographic routing protocol. The results of GeoNet project are presented in the paper, which aims at combining IPv6 networking and C2CNet. The system with IPv6 and C2CNet is designed and implemented in Linux. The prototype implementation is first evaluated indoor testbed with the fixed positions. Then it is evaluated in the field testbed with three vehicles with various scenarios. For evaluation in field testbed, we have developed the AnaVANET evaluation tool to perform the evaluation taking into account all of geographic factors.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Experiment

Keywords

VANET, Geographic routing, IPv6

1. INTRODUCTION

Humans spend considerable time in the vehicle these days. ITS is going to be more and more important technologies in our life, that enhance safety, driving efficiency and amusing by allowing various service such as fleet management, navigation, billing multimedia application and game. IPv6 is considered as the most appropriate technologies to support communication in ITS thanks to its extended address space, embedded security, enhanced mobility support and ease of configuration. Future vehicles will embed a number of sensors and other devices that could be IPv6 enabled [1].

In vehicular networks, vehicles equip with On-Board Units (OBUs) to enable the communication with other vehicles. Vehicle-to-vehicle ad hoc networks are multihop communication using geographic position, which has been investigated on GeoNet Project [2]. On the other hand, Roadside Units (RSUs) are installed around the road. IEEE802.11 is used to connect between OBUs, and between OBU and RSU. Application Unit (AU) is a portable or built-in device connected temporarily or permanently to the vehicle's OBU. OBU also can be connected to the Internet with cellular networks, WiMAX, etc. These terminologies are proposed in Car2Car communication consortium (C2C-CC [3]).

The organization of the paper is as follows: Section 2 describes the GeoNet approach to make IPv6 work over C2CNet that specified in C2C-CC. Section 3 shows the overview of prototype implementation on Linux system. Section 4 evaluates network performance using the implementation with indoor testbed and Section 5 shows evaluation with vehicles. Section 6 concludes the paper and shows the future works.

2. ENABLING IPV6 OVER C2CNET

In this section, we first describe the C2CNet and objective of the GeoNet project. Then, the how to interact between IPv6 and C2CNet is described. The specification of the interface between IPv6 and C2CNet is called the IP-C2C Service Access Point (SAP) in the GeoNet project.

2.1 C2CNet and GeoNet project

C2C-CC is designing a separate network protocol (C2CNet) different from Internet Protocol (IP) to ensure car-to-car communication for both safety and non-safety and with taking into consideration both availability and non-availability of infrastructure. C2CNet protocol is tailored for vehicular environments and would rely on position-based routing. This protocol would define a separate C2CNet header with a separate C2CNet identifier, tentatively 64-bit length, identifying C2CNet node. C2CNet header is planned to carry source C2CNet identifier, destination C2CNet identifier, source geographic location and destination geographic location.

The GeoNet project started from February 2008 as an European project and aims at combining IPv6 networking and C2CNet. In C2C-CC architecture, C2CNet layer is located between IPv6 and link layers. Thus IPv6 packet is delivered with outer C2CNet header as depicted in Figure 1. The challenge is how to support the communication types defined in C2CNet in IPv6 layer. The objective of GeoNet is to improve these specifications and create a prototype software implementation interfacing with IPv6. The goal of GeoNet is thus to implement and formally test a networking mechanism as a standalone software module which can be incorporated into Cooperative Systems.

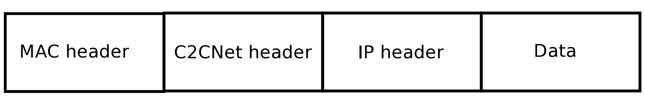

Figure 1: Packet encapsulation

2.2 Service Access Point of IPv6 and C2CNet

The Service Access Point (SAP) between IPv6 and C2CNet is the interface to transmit the packet up from the C2CNet layer to IP layer and down from IP layer to C2CNet layer. Four types of communication are defined in C2CNet architecture: GeoUnicast, GeoBroadcast, GeoAnycast and TopoBroadcast. First three are the type of communication which based on geographic information and the last one is based on network topology information. *GeoUnicast* routes data from a source node to a destination node for which the exact geographical location is known. *GeoBroadcast* delivers data from a source node to all nodes located within a specific geographical area. And *GeoAnycast* routes data from a source node to any node located within a specific geographical area. *TopoBroadcast* routes data from a source node to all nodes located up to a specific distance in terms of hops.

To enable these communication types with IPv6, we decide to use unicast, multicast, anycast in IPv6 layer as shown in table 1.

According to Table 1, only one function, named GeoIPv6, is defined to transmit the packet from IP layer to C2CNet layer. In this function three parameters could be considered: scope, destination and payload.

Table 1: Types of destinations

Destination	C2CNet	IPv6
A node in a specific car	GeoUnicast	unicast
Nodes in cars in area	GeoBroadcast	multicast
Nodes in cars x hops away	TopoBroadcast	multicast
A node in a car in area	GeoAnycast	anycast

- **Scope**: according to the destination type as described in Table 1, four scopes are needed: GeoUnicast, GeoAnycast, GeoBroadcast and TopoBroadcast. These correspond to IPv6 unicast, IPv6 anycast, and IPv6 multicast packets, respectively.
- **Destination**: In unicast, IP layer provides, to C2CNet layer, IP next hop as destination address and GeoRouting module determines C2CNet ID from IP next hop. On the other hand, in the case of GeoBroad cast and GeoAnycast, GeoDestination ID is provided to C2CNet layer. For circle area, the center position (latitude and longitude) and radius is resolved in C2CNet layer by GeoDestination ID. For TopoBroadcast, hop limit is provided from IP to C2CNet.
- **Payload**: contains IP packet.

3. DESIGN AND IMPLEMENTATION

The prototype system is implemented on GNU/Linux (kernel 2.6.29). In this section, design and implementation are mentioned.

3.1 System design

The C2CNet functions are divided into three main modules that cooperate each other. The three modules are implemented in userland for ease of implementation and modification. Remind that one of objective of the project is to brush the specification up by feedback from the implementation. The three modules are responsible of particular function on OBU. These modules cooperate via inter-process communication socket.

PositionSenser module is to create a stable interface for acquiring geographic data by the C2CNet modules. It is implemented as a stand-alone program connected to a positioning service available for a particular platform. It sends the position information over a UDP socket to C2CNet modules. **Lowerlayer** module is the interface between C2CNet (GeoNet internal modules) and the PHY/MAC Layer. This is needed to support the platform independency of GeoNet. It allows GeoNet to support different platforms with different network-interfaces without holding platform specific parameters within the C2CNet modules. **C2CNet** module controls the position information and keeps transmitting a periodic packet to inform its neighbors about its presence. It also transmits received data with C2CNet header to Lowerlayer module via UDP socket. IP-C2C SAP, which has GeoIPv6 function described in Section 2.2, is integrated into C2CNet module.

3.2 Overview of the system

In Linux system, IPv6 packet forwarding is processed in the kernel space. However the packet has to be brought to the user land from kernel, because the C2CNet module is implemented in userland. Then the packet is encapsulated

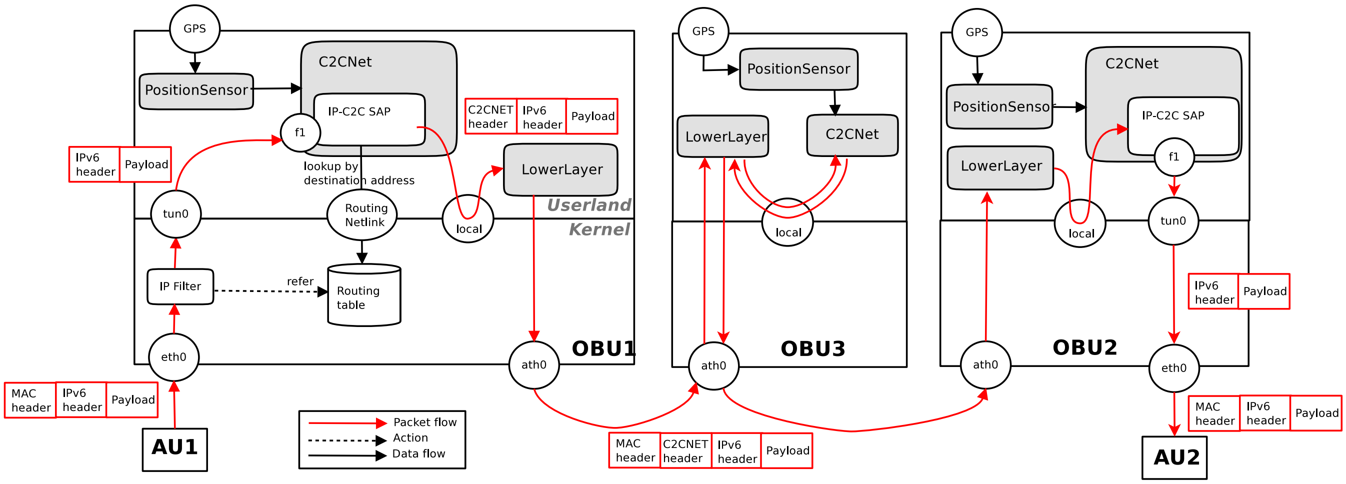

Figure 2: Implementation of IPv6 over C2CNet

with C2CNet header and then sent back to the kernel again. We decide to use TUN virtual interface to bring the packet to the user land. Overall process of IPv6 over C2CNet is illustrated in Figure 2.

AU1 sends IPv6 packets to OBU1 that is the default router of in-vehicle network. OBU1 receives the packets on the ingress interface (eth0 in Figure 2) and removes MAC header of the packets. Then IP header and payload part are transmitted into the tun0 virtual interface by the pre-configured rules of IP Filter [1]. The C2CNet module reads the data from tun0 and parses the information of the IP header.

The destination IPv6 address is used to distinguish communication type whether unicast or multicast by the first 8 bits which are correspondent to GeoUnicast and GeoBroadcast, respectively. In unicast case, the next hop IPv6 address is resolved from the routing table via netlink library by the destination IPv6 address. The last 64-bits of the next hop IPv6 address is correspondent to the destination C2CNet ID. In multicast case, destination C2CNet information are pre-configured depending on the destination IPv6 address (i.e. if the destination address is link-local all node multicast address (ff02::1), the latitude and longitude are as well as those of OBU1 and the radius is 500 meter).

The data with C2CNet header, IPv6 header and payload are sent to LowerLayer module via local UDP socket. LowerLayer module adds MAC header over C2CNet header and transmits the frame into the air. The intermediate node (OBU3) receives the frame and re-transmits the frame when C2CNet modules find that the frame should be re-transmitted to reach the destination with multihop manner.

Finally, OBU2 receives the frame and on the egress interface. Then Lowerlayer module removes the MAC header. And C2CNet module finds that the destination of the C2CNet packet is OBU2. The IPv6 header and payload are sent to the tun0 virtual interface. The packet is routed to egress interface (eth0). And AU2 receives the IPv6 packet that sent from AU1.

4. INDOOR EVALUATION

4.1 Testbed Configuration

Network performances were, first, evaluated in the indoor testbed to avoid interferences due to unexpected radio perturbations and difficulties to trace the movements of the OBUs. The following experiments were performed outside of any vehicles. Both the OBUs and AUs did not actually move during the experiments. The network configuration is same as in Figure 2 with single hop and multiple hop environment. We tested the Round Trip Time (RTT) using ICMPv6 and the packet delivery ratio and the bandwidth using UDP with various parameters.

4.2 Latency measurement

To measure Round Trip Time (RTT) between AUs, AU1 sent ICMPv6 packet 100 times in 10 seconds (interval 0.1 seconds). The packet size is increased by 20 bytes each 10 seconds from 20 bytes until 1500 bytes. There was no traffic other than ping6. Figure 3 shows the result.

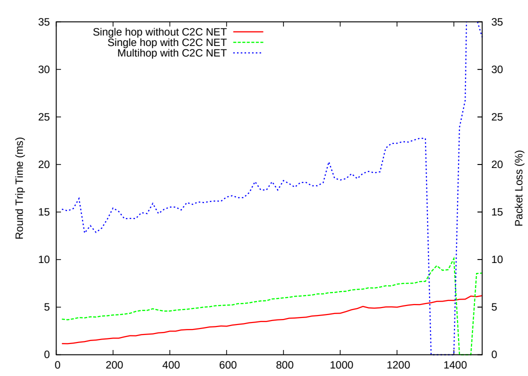

Figure 3: RTT between AUs

It shows the RTT on single hop without C2CNet (red line), single hop with C2CNet (green line) and multi-hop with C2CNet (blue line). In the single hop case, the RTT

[1] http://www.netfilter.org

with C2CNet is 3 ms higher than one without C2CNet. In addition, packets with size exceeding 1300 bytes cannot be delivered with C2CNet because of the MTU, while the packet without C2CNet is delivered until 1500 bytes.

4.3 Packet delivery ratio and Bandwidth

In the test, UDP packets are sent from AU1 to AU2 during 20 seconds by *iperf* command. The sender sent from 1Mbits/sec to 6Mbits/sec with various size of packet from 100 bytes to 1900 bytes. There was no traffic other than iperf traffic. Figure 4 shows the result.

Figure 4 shows the packet delivery ratio on single hop. Packet delivery ratio is low while packet size is small. There is no packet loss with a packet size between 700 bytes and 1300 bytes with 1M of sending rate, between 900 bytes and 1300 bytes of packet size with 2M sending rate and between 1100 bytes and 1300 bytes of packet size with 3M sending rate.

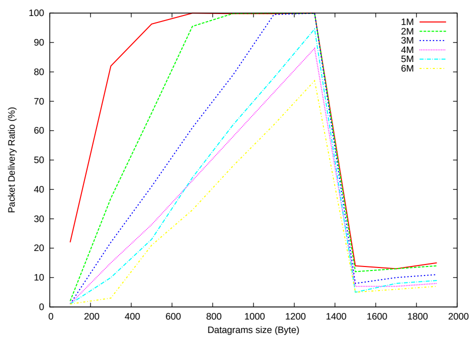

Figure 4: Packet delivery ratio on single hop with UDP

Figure 5 shows the throughput for the same tests as reported on Figure 4. The throughput is maximized with a 1300 bytes packet size for all sending rates. It shows that the most efficient configuration to send maximum data is realised with a 1300 bytes packet size and 5M sending rate. Maximum throughput is around 4500 Kbits/second.

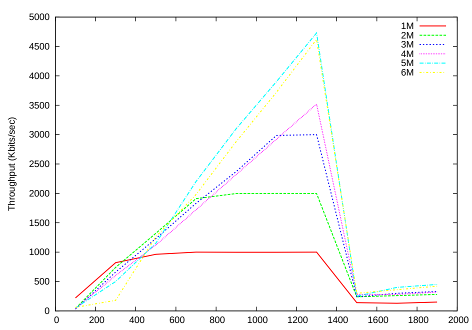

Figure 5: Throughput on single hop with UDP

5. FIELD EVALUATION

5.1 Equipments

To evaluate the performance in more realistic scenarios, we setup an outdoor field test environment with three vehicles equipped with an OBU, an AU, GPS receiver and wifi antenna as shown in Figure 6. OBUs are Alix3d3 embedded PCs on which Ubuntu 9.0.4 is installed with a Linux 2.6.29.6 kernel. Each OBUs has one built-in Ethernet port (ingress interface) which is connected to the Ethernet hub connecting other PCs, and a mini-pci wireless card (Atheros AR5414 802.11 a/b/g Rev 01) used as wireless connection to other OBUs. The OBUs are also connected, via serial port, with a Trimble AgGPS 323 GPS receiver, whose external antenna is visible in the photo.

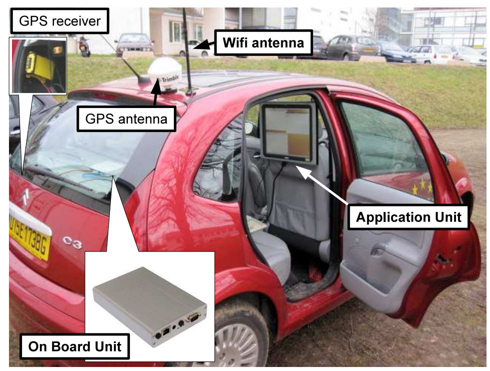

Figure 6: Equipments

5.2 Evaluation tool: AnaVANET

The topology of the network dynamically changes during the test depending on the location of the vehicles. The performance of IPv6 GeoNetworking depends on the radio propagation which is influenced by obstacles. Network performance also depends on other factors such as the distance, movement of vehicles. We have therefore developed the AnaVANET evaluation tool to perform the evaluation taking into account all of these factors.

AnaVANET is a tool developed internally at INRIA to analyze vehicular networks. It has originally been used to evaluate OLSR-based ad-hoc vehicular networks [4]. For the purpose of evaluating the performance of IPv6 GeoNetworking, AnaVANET is extended in order to analyze IPv6 packets transmitted with a C2CNet header in the GeoNet domain.

Figure 7 provides an overview of the experimental evaluation process carried out in the tests. The Sender (AU) is in charge of generating data traffic, and both the sender and the receiver save a high level log, according to the application used to generate network traffic. All OBUs record information about forwarded data packets by means of the *tcpdump* software, and log the vehicle position continuously. All this data is post-processed by the AnaVANET software and then analyzed. A Java application traces all the data packets transmitted from the sender node. This way, it is possible to detect packet losses and calculate statistics for each link and end-to-end, and merge all these per-hop information with transport level statistics of the traffic generator. As a result, AnaVANET outputs an XML file with statistics of one-second periods, and a packet trace file with the path followed by each data packet.

The XML file is compatible with google maps API and web based analysis is produced. The experiments carried out are available on the GeoNet [2] web site and can be replayed to see the momentary performance of the network during the tests. All the experiments can be selected and main performance metrics can be monitored at any time. From the two types of output file from AnaVANET,the *gnuplot* software generate the graphs that also appear in rest of the paper.

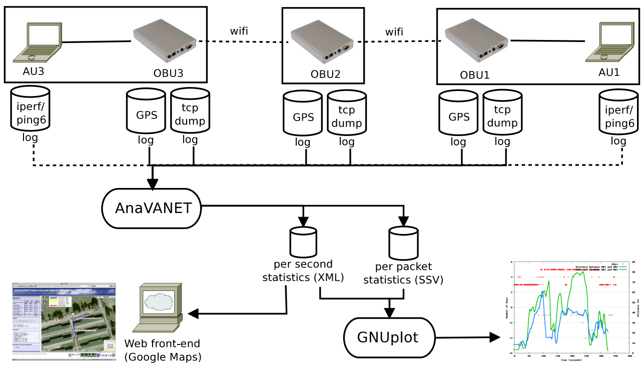

Figure 7: Overview of AnaVANET

5.3 Maximum Range Tests

The test was performed with one parked vehicle and one moving vehicle, when vehicles were in normal urban conditions and isolated from obstacles and interferences. The aim of this scenario was to check the maximum distance the wireless range can reach.

The packet delivery ratio using UDP with this scenario is shown in Figure 8. The packet delivery ratio is almost 100 % from beginning to 200 meters. From 200 meters, the packets are starting to be dropped and the packet transmission finally ends at a distance of 420 meters. The packets are not delivered until the vehicle comes back to a distance of 400 meters 50 seconds after the communication ends.

The jitter of in the same test is illustrated on Figure 9. When the sender car leaves the receiver one, at a distance between 250 and 420 meters, the jitter is higher, due to layer two retransmissions caused by the increase of the distance. When the sender approaches the receiver again, this effect, but higher, is again visible at distances between 400 and 200 meters. This is due to packet flood of buffered packet during the disconnected period.

Figure 10 shows the RTT with ICMPv6 transmission. The RTT is within 5 ms to 10 ms until 420 meters. After this point, no packets are delivered, until the sender vehicle comes back and reaches 100 meters of distance. Since periodical C2CNet beacon messages are lost when the distance is around 420 meters, the destination C2CNet ID is removed from the location table and the transmission ends at this point.

Throughput using TCP and considering the same scenario is given in Figure 11. The maximum throughput is around 1000 Kbits/sec when the vehicles are parked next to one another. When the distance is from 50 meters to 200 meters, the average throughput is around 500 Kbits/sec and

[2]http://www.geonet-project.eu/demonstration/geonet/

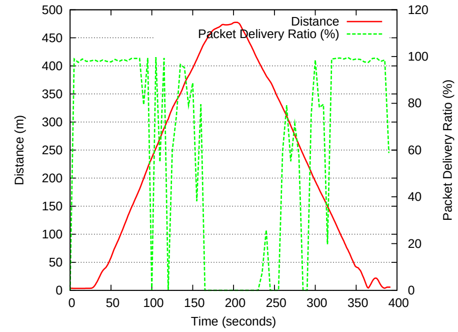

Figure 8: Distance with UDP (distance/PDR)

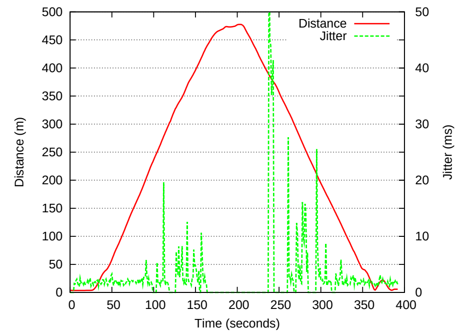

Figure 9: Distance with UDP (distance/jitter)

the TCP communication is interrupted at 270 meters. The communication does not recover during the rest of the test, because the TCP session time out. It takes 50 seconds to come back to a distance of 100 meters where ICMPv6 recovered during previous test.

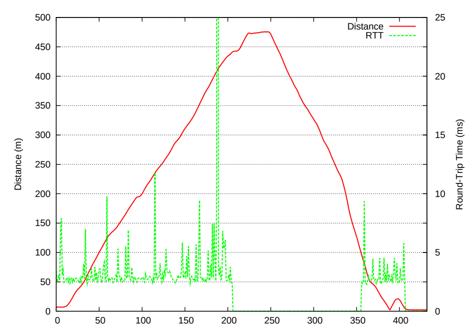

Figure 10: Distance with TCP (distance/RTT)

5.4 Dynamic test up to 30 Km/h

Figure 12 shows hop count, packet delivery ratio and jitter on dynamic tests under 30 km/h. The upper plot shows the number of hops used in the paths followed by UDP packets, whereas the lower graphs show the packet delivery ratio, computed end to end and per link. The packet delivery ratio is calculated per second, while the hop-count is plotted for each packet transmitted from the sender node. When no hops are drawn, the route to the destination vehicle is not

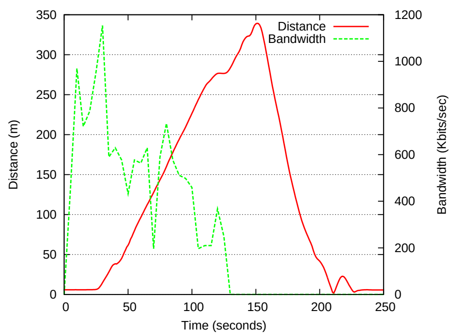

Figure 11: Distance with TCP (distance/bandwidth)

available.

Zero hops means that the packet was sent by the first OBU but was not received by any other. Negative values represent those packets which did not arrive to the destination vehicle, but some hops were reached. As can be seen, a direct relation exists between packet delivery ratio and number of hops. When this last value is equal or lower than zero, the packet delivery ratio decreases. When the vehicles are in the same street, some direct paths (one-hop) appear; however, when the distance between the sender and the receiver vehicles is large enough, the two-hop route is used. These different types of paths can be also seen if the per-link packet delivery ratio is observed. Whereas the direct link (OBU3-OBU1) gives intermediate packet delivery ratio values, the packet delivery ratio between consecutive vehicles is almost identical and near 100 % when the two-hop link is used, due to the lower distance between nodes.

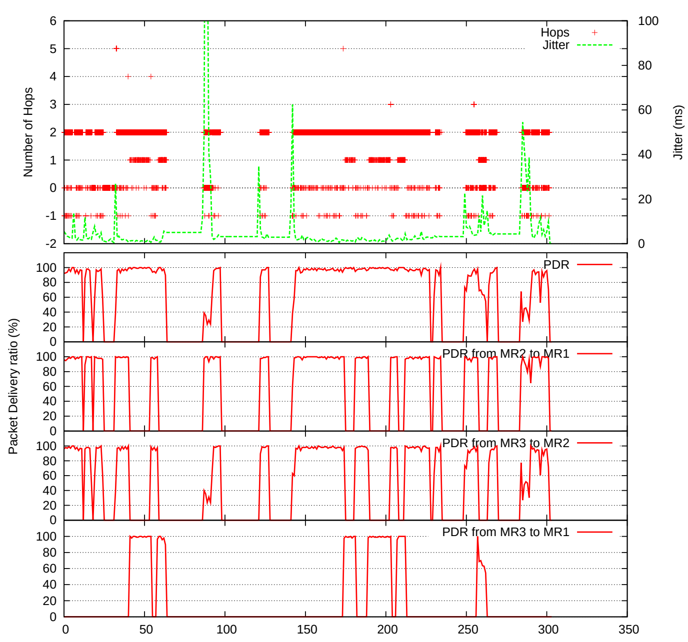

Figure 12: UDP test with Dynamic test up to 30 Km/h

6. CONCLUSION AND FUTURE WORKS

We shows that how to enable IPv6 networking over C2CNet which is specified in Car2Car Communication Consortium as a geographic routing protocol. Then the system is divided into three functionalities and implemented as three modules in Linux.

We have set up an experimental indoor testbed and outdoor testbed to investigate the network performance on IPv6 over C2CNet. The indoor test environment is designed to evaluate the pure performance of IPv6 over C2CNet avoiding interferences due to unexpected radio perturbations. We measured the network performance with UDP and ICMPv6 traffic using iperf and ping6. The test results show that IPv6 over C2CNet does not have too much delay (less than 4ms with a single hop) and is feasible for vehicle communication. In the outdoor testbed, we developed AnaVANET to enable hop-by-hop performance measurement and position trace of the vehicles.

We focused on Vehicle-to-Vehicle scenarios in the reported evaluation but we also intend to continue the evaluation with Vehicle-to-Infrastructure scenarios (Roadside- based and/or Internet-based. GeoNet OBUs comprise NEMO [5] and MCoA [6] functionalities. We are motivated to measure the network performance using these functionalities over C2CNet.

The combination of IPv6 multicast and GeoBroadcast was implemented, however we could not evaluate the performance with such a scenario. One of the reasons is that a sufficiently high number of receivers is necessary to properly evaluate multicast but experimental evaluation is limited in the number of vehicles (4 in our case). So evaluation of IPv6 GeoNetworking with multicast capabilities by means of simulation or emulation is thus left for future work.

In the evaluation, we only tested UDP, TCP, and ICMPv6 with fixed sending rate. The performance of an actual traffic hazard application such as the one used in the final GeoNet workshop is still not evaluated. In the future, the performance of IPv6 GeoNetworking should be evaluated under more realistic scenario such as this one.

7. REFERENCES

[1] T. Ernst. The Information Technology Era of the Vehicular Industry. *ACM SIGCOMM Computer Communication Review (CCR)*, Volume 36(Issue 2), April 2006.

[2] Geonet project: http://www.geonet-project.eu.

[3] Car-to-car communication consortium: http://www.car-to-car.org.

[4] J. Santa, M. Tsukada, T. Ernst, and A. F. Gomez-Skarmeta. Experimental analysis of multi-hop routing in vehicular ad-hoc networks. In *Proc. 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks. Communities and Workshops TridentCom 2009*, pages 1–8, April 6–8, 2009.

[5] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. *Network Mobility (NEMO) Basic Support Protocol*, January 2005. IETF RFC3963.

[6] R. Wakikawa, V. Devarapalli, G. Tsirtsis, T. Ernst, and K. Nagami. *Multiple Care-of Addresses Registration*, November 2009. RFC 5648.